

What is claimed is:

1. 1. A method of building distributed software services as aggregations of other services, comprising steps of:
 3. determining a taxonomy of interest for a new distributed software service;
 4. programmatically scanning a network-accessible registry to locate registered services
 5. having the taxonomy of interest;
 6. determining if each located service has a predetermined deployment interface; and
 7. providing the located services to a service composition tool if the determining step has a positive result.
2. 2. The method according to Claim 1, further comprising the step of adding the predetermined deployment interface to the located service when the determining step has a negative result.
3. 3. The method according to Claim 2, wherein the adding step causes the determining step to have the positive result.
2. 4. The method according to Claim 1, further comprising the steps of:
 2. determining if each located service has a predetermined system interface for managing
 3. that located service.
1. 5. The method according to Claim 4, further comprising the step of adding the

2 predetermined system interface to the located service when the determining step has a negative
3 result.

1 6. The method according to Claim 1, wherein the providing step further comprises the step
2 of placing a representation of the located services on a user interface of the composition tool..

1 7. The method according to Claim 1, further comprising the step of building the new service
2 using selected ones of the provided services.

1 8. The method according to Claim 7, further comprising the step of obtaining information
2 regarding the provided services for use in the building step.

1 9. The method according to Claim 8, wherein the obtained information comprises
2 operations available from the provided services.

1 10. The method according to Claim 8, wherein the obtained information comprises an author
2 of each of the provided services.

1 11. The method according to Claim 8, wherein the obtained information comprises a
2 descriptive name of each of the provided services.

1 12. The method according to Claim 8, wherein the obtained information comprises an iconic

2 representation of each of the provided services.

1 13. The method according to Claim 7, further comprising the step of creating a directed graph
2 representation of the new service.

1 14. The method according to Claim 13, wherein nodes of the directed graph represent
2 operations of the new service and edges of the directed graph represent transition conditions of
3 the new service.

15. The method according to Claim 14, wherein data mapping operations are associated with
selected ones of the edges.

16. The method according to Claim 13, further comprising the step of creating a markup
language document representing the directed graph representation.

1 17. The method according to Claim 16, wherein the markup language document is a Web
2 Services Flow Language (“WSFL”) document.

1 18. The method according to Claim 16, further comprising the step of registering the markup
2 language document in a network-accessible registry.

1 19. The method according to Claim 18, wherein the registered markup language document
2 thereby becomes available for locating in subsequent iterations of the programmatically scanning
3 step.

1 20. The method according to Claim 7, wherein the new service is modeled as a portlet.

1 21. The method according to Claim 7, further comprising the step of defining a public
2 interface to the new service.

1 22. The method according to Claim 7, further comprising the step of defining a deployment
2 interface to the new service.

1 23. The method according to Claim 7, further comprising the step of defining a system
2 interface to the new service.

1 24. The method according to Claim 7, further comprising the step of selecting one or more
2 service providers to fulfill operations of the new service.

1 25. The method according to Claim 24, further comprising the steps of:
2 determining a selected taxonomy wherein service providers may be located for binding to
3 the new service;
4 determining a particular network-accessible registry wherein the service providers may be

5 registered; and

6 programmatically scanning the particular registry to locate service providers having
7 registered services in the selected taxonomy.

1 26. The method according to Claim 25, wherein the programmatically scanning step further
2 comprises determining if the registered services of the located service providers have a
3 predetermined deployment interface.

1 27. The method according to Claim 24, further comprising the step of defining a mapping
2 between operations of the selected service providers and operations of the new service.

28. The method according to Claim 27, wherein the mapping comprises transformation logic.

29. The method according to Claim 27, wherein the mapping comprises stylesheet
transformations.

1 30. The method according to Claim 28, wherein the transformation logic specifies service-to-
2 service transformations for particular services.

1 31. The method according to Claim 24, further comprising the step of creating a static
2 binding to the selected service providers.

1 32. The method according to Claim 24, further comprising the step of creating a dynamic
2 binding to the selected service providers.

1 33. The method according to Claim 32, wherein the dynamic binding includes syntax to
2 programmatically select particular service providers at run-time.

1 34. The method according to Claim 27, further comprising the steps of:
2 determining that no mapping exists; and
3 prompting a composer of the new service to identify a source of the mapping.

1 35. The method according to Claim 31, further comprising the step of creating a markup
2 language document representing the binding.

1 36. The method according to Claim 35, further comprising the step of registering the markup
2 language document in the registry.

1 37. The method according to Claim 32, further comprising the step of creating a markup
2 language document representing the binding.

1 38. The method according to Claim 37, further comprising the step of registering the markup
2 language document in the registry.

1 39. The method according to Claim 38, wherein the markup language document is a Web
2 Services Flow Language (“WSFL”) global model.

1 40. The method according to Claim 1, wherein the new service is a web service.

1 41. A system for building distributed software services as aggregations of other services,
2 comprising:

3 means for determining a taxonomy of interest for a new distributed software service;

4 means for programmatically scanning a network-accessible registry to locate registered
5 services having the taxonomy of interest;

6 means for determining if each located service has a predetermined deployment interface;

7 and

8 means for providing the located services to a service composition tool if the determining
9 step has a positive result.

1 42. The system according to Claim 41, further comprising means for adding the
2 predetermined deployment interface to the located service when the means for determining has a
3 negative result.

1 43. The system according to Claim 41, further comprising means for determining if each
2 located service has a predetermined system interface for managing that located service.

1 44. The system according to Claim 41, further comprising means for building the new service
2 using selected ones of the provided services.

1 45. The system according to Claim 44, further comprising means for creating a directed
2 graph representation of the new service, wherein nodes of the directed graph represent operations
3 of the new service and edges of the directed graph represent transition conditions of the new
4 service, and wherein data mapping operations are associated with selected ones of the edges.

1 46. The system according to Claim 45, further comprising means for creating a markup
2 language document representing the directed graph representation and registering the markup
3 language document in a network-accessible registry.

1 47. The system according to Claim 44, wherein the new service is modeled as a portlet, and
2 further comprising means for defining a public interface to the new service.

1 48. The system according to Claim 44, further comprising means for defining at least one of
2 (1) a deployment interface to the new service and (2) a system interface to the new service.

1 49. The system according to Claim 44, further comprising means for selecting one or more
2 service providers to fulfill operations of the new service.

1 50. The system according to Claim 49, further comprising:

2 means for determining a selected taxonomy wherein service providers may be located for
3 binding to the new service;
4 means for determining a particular network-accessible registry wherein the service
5 providers may be registered; and
6 means for programmatically scanning the particular registry to locate service providers
7 having registered services in the selected taxonomy.

1 51. The system according to Claim 49, further comprising means for defining a mapping
2 between operations of the selected service providers and operations of the new service, wherein
3 the mapping comprises at least one of (1) transformation logic, (2) stylesheet transformations,
4 and (3) service-to-service transformations for particular services.

2 52. A computer program product for building distributed software services as aggregations of
3 other services, the computer program product embodied on one or more computer-readable
4 media and comprising:

4 computer-readable program code means for determining a taxonomy of interest for a new
5 distributed software service;
6 computer-readable program code means for programmatically scanning a network-
7 accessible registry to locate registered services having the taxonomy of interest;
8 computer-readable program code means for determining if each located service has a
9 predetermined deployment interface; and
10 computer-readable program code means for providing the located services to a service

11 composition tool if the determining step has a positive result.

1 53. The computer program product according to Claim 52, further comprising computer-
2 readable program code means for adding the predetermined deployment interface to the located
3 service when the computer-readable program code means for determining has a negative result.

1 54. The computer program product according to Claim 52, further comprising computer-
2 readable program code means for determining if each located service has a predetermined system
3 interface for managing that located service.

1 55. The computer program product according to Claim 52, further comprising computer-
2 readable program code means for building the new service using selected ones of the provided
3 services.

1 56. The computer program product according to Claim 55, further comprising computer-
2 readable program code means for creating a directed graph representation of the new service,
3 wherein nodes of the directed graph represent operations of the new service and edges of the
4 directed graph represent transition conditions of the new service, and wherein data mapping
5 operations are associated with selected ones of the edges.

1 57. The computer program product according to Claim 56, further comprising computer-
2 readable program code means for creating a markup language document representing the directed

3 graph representation and registering the markup language document in a network-accessible
4 registry.

1 58. The computer program product according to Claim 55, wherein the new service is
2 modeled as a portlet, and further comprising computer-readable program code means for defining
3 a public interface to the new service.

1 59. The computer program product according to Claim 55, further comprising computer-
2 readable program code means for defining at least one of (1) a deployment interface to the new
3 service and (2) a system interface to the new service.

1 60. The computer program product according to Claim 55, further comprising computer-
2 readable program code means for selecting one or more service providers to fulfill operations of
3 the new service.

1 61. The computer program product according to Claim 60, further comprising:
2 computer-readable program code means for determining a selected taxonomy wherein
3 service providers may be located for binding to the new service;
4 computer-readable program code means for determining a particular network-accessible
5 registry wherein the service providers may be registered; and
6 computer-readable program code means for programmatically scanning the particular
7 registry to locate service providers having registered services in the selected taxonomy.

1 62. The computer program product according to Claim 60, further comprising computer-
2 readable program code means for defining a mapping between operations of the selected service
3 providers and operations of the new service, wherein the mapping comprises at least one of (1)
4 transformation logic, (2) stylesheet transformations, and (3) service-to-service transformations
5 for particular services.